GVP Deployment Guide

How Logging and Reporting Works

5/3/2025

How Logging and Reporting Works

GVP Reporting refers to the GVP logging and reporting feature, which provides these features and services:

- Accumulating key measurements and data that describes the calls being processed by the deployment
- Infrastructure capable of reliable delivery of data to a relational back end
- Near real-time reporting about operational aspects of the deployment
- Historical reporting about VoiceXML and CCXML application usage

For an overview of the GVP Reporting architecture, see GVP Reporting Architecture. Read here about:

GVP Logging

The GVP Logging API enables GVP components to raise logging events at two levels:

- *logs* are events at the level of the component, or GVP Application object.
- *metrics* are events at the level of the VoiceXML or CCXML application.

Logs

Logs include three important elements by which they can be filtered: severity, module ID, and specifier.

Severity

Like most other Genesys components, GVP components can raise events at various levels, however, they can also log them further by the following levels of severity (in order of descending severity):

- 0 = Critical
- 1 = Error
- 2 = Warning
- 3 = Note
- 4 = Info
- 5 = Debug

Module IDs

Each GVP component is composed of one or more application modules, each of which is assigned a Module ID. The component logically organizes the logs that it emits by Module ID.

Specifiers

A specifier is a number that uniquely identifies a given event that is logged by a given module.

For a list of the Module IDs and specifiers that are used in GVP 9.0, see the Genesys Voice Platform 8.5 User's Guide.

Additional Log Data

GVP Logging associates a UTC timestamp, to millisecond precision, with each logging event when it is raised.

Logs for call-processing component events that are associated with a call session include the GVP component ID (DBID of the GVP application that raised the log), the GVP session ID, and the UTC timestamp. Metrics can therefore be mapped to CDRs, which provide more information, such as the call start time, call end time, IVR Application ID (DBID of the IVR Profile), Tenant ID, and local and remote SIP URIs.

The GVP session ID is also unique to the deployment, where the environment may include deployments at multiple sites or geographic locations.

For more information about GVP IDs, see the section about GVP identifiers in the Genesys Voice Platform 8.5 User's Guide.

Log Delivery

Log events are delivered to one or more log sinks for the component (see Log Sinks), and then sent to Management Framework or the Reporting Server.

By default, log files are stored in the following location:

```
C:\Program Files\GCTI\gvp\<IP name>\<Your component application name>\logs\
```

The Management Framework Adapter sink passes along GVP log messages to its own logging system. In general, GVP logging is mapped to the Management Framework logging levels in the following way:

- Metrics are mapped to the interaction level and have IDs between `50000-55000`.
- Critical, Error, and Warning logs are mapped to the standard level.
- Notice and Info logs are mapped to the trace level.
- Debug logs are mapped to the debug level and usually have an ID of `20000`.

For a list of Management Framework IDs for GVP log messages, see the LMS file for each GVP component. The LMS file contains a mapping of GVP log messages to Management Framework IDs. The LMS file is located in the `<Install_Dir>\config` directory for each GVP component.

For more information about the GVP log messages, see the appendix "*Module and Specifier IDs*" in the Genesys Voice Platform 8.5 User's Guide.

Metrics

Metrics describe application-level events, and have no severity.

Each metric has a unique type identifier (for example, start_session) and is associated with a specific VoiceXML or CCXML session ID. The body of the metric is defined by the component. For example, the body of a metric can be a text string that consists of a number of pipe-delimited parameters (such as `ANI|DNIS|SIP Request-URI`) encoded in UTF-8.

Metrics Examples

The following are examples of the kinds of metrics that are logged. For full details about the metrics that are available in GVP 9.0, see the Genesys Voice Platform 8.5 Metrics Reference Guide.

- The Media Control Platform logs an INCALL_BEGIN metric when an inbound call is accepted.
- The NGI logs a PROMPT metric when it starts to play back a prompt queue.
- The amount of time to fetch a VoiceXML page is measured and logged.

VAR Metrics

Voice Activated Response (VAR) metrics are events that the Media Control Platform generates when it encounters VAR-specific <log> tags in the VoiceXML applications. The VAR-specific `<log>` tags have the prefix `com.genesyslab.var`.

For the metrics that the Media Control Platform generates when the NGI executes a VAR-specific `<log>`; tag, see the Media Control Platform reference information appendix in the Genesys Voice Platform 8.5 User's Guide.

For more information about using `<log>` tags in VoiceXML applications, see the Genesys Voice Platform 8.1 Genesys VoiceXML 2.1 Reference Help.

Metrics Delivery

Metrics are delivered to the log sinks for the component (see Log Sinks). Upstream metrics, which are also referred to as call events, are metrics that are configured to be sent to the Data Collection Sink (DATAC), and then to the Reporting Server for storage and reporting purposes. The DATAC also computes service-quality measurements based on the logs and metrics that are forwarded to it. The Media Control Platform and Call Control Platform are the only sources of upstream metrics.

Log Sinks

Every component that uses logging has configurable access to one or more log sinks, that receive a real-time stream of logs or metrics, as defined by filters that you can configure (in the `ems.logconfig.<`*`Sink Name`*`>` and `ems.metricsconfig.<`*`Sink Name`*`>` parameters).

The log sinks enable GVP Reporting to implement upstream reporting, integrate with Management Framework, and accumulate summary statistics that are used by the Reporting Server. Upstream reporting refers to the ability of components to send a configured subset of metrics to the Reporting Service for storage and reporting purposes.

The following log sinks are available in GVP:

- **MFSINK**—The Management Framework Adaptation Sink. MFSINK connects the GVP and Management Framework logging systems, through CCILib, for file-based and network-based logging. Configurable parameters in the log configuration section for each GVP component determine what Management Framework does with the logs for example, writing them to file or delivering them to Message Server.
- **DATAC**—The Data Collection Sink. DATAC derives resource-specific summary statistics, and delivers summary statistics and metrics to the Reporting Server, where they can be queried through the Call Events reporting service. (This sink is not applicable for the Resource Manager or Fetching Module.) The `ems.dc.default.metricsfilter` configurable option on the Media Control Platform and Call Control Platform enables you to specify which of the metrics delivered to DATAC will be forwarded to the Reporting Server.
- **TRAPSINK**—The SNMP integration sink. For GVP components that have been configured to raise traps, TRAPSINK forwards log messages to the Management Data Agent library, which the GVP process uses to implement the applicable management information bases (MIB).

The log sinks are dynamic link libraries (DLL) that are loaded dynamically at runtime. By default, the following log sinks are attached to each component:

- Resource Manager and Fetching Module MFSINK and TRAPSINK.
- Media Control Platform and Call Control Platform DATAC, MFSINK, and TRAPSINK.

Depending on the configured filters, a particular log or metric may be directed to more than one log sink for the component, or to none. If a given log event does not match the configured event types for that component s log sinks, the log event is silently discarded.

The log sinks themselves can generate log events, therefore, they have one or more log sinks attached to them. For example, starting in GVP 8.1, DATAC has an MFSINK, which enables DATAC logs to be delivered to Message Server or to file.

CDR Reporting

Call-detail records are records that describe key attributes of a call session that the deployment is processing, or has processed.

The CDR Service on the Resource Manager, Media Control Platform, and Call Control Platform enables the component to submit and update CDRs to the Reporting Server in near-real time.

The Reporting Server correlates the CDRs based on the GVP Session-ID. This is the ID that the Resource Manager assigns to all calls that come into GVP. For more information about GVP session-IDs, see the Genesys Voice Platform 8.5 User's Guide.

The intervals at which the Reporting Client submits CDRs to the Reporting Server depends on the configuration. For configuration considerations, see the description of the ems.rc.cdr.batch_size configuration parameter in the section about configuring GVP Reporting in the Genesys Voice Platform 8.5 User's Guide.

CDR Attributes

The CDRs share a common set of attributes that, at a minimum, the component must include. In addition, the Media Control Platform and Call Control Platform include certain attributes that are specific to the component type.

Common CDR Attributes

All components include the following attributes in the CDRs that they submit:

- Session start and end times.
- IDs for the VoiceXML or CCXML application; Media Control Platform or Call Control Platform session; Resource Manager session; and overall Genesys session (UUID).
- IDs for the tenant for which the call is associated. The CDR report returns CDRs for all tenants that are in the tenant hierarchy.
- Call type Available types are:
  - `Inbound (1)` For the Resource Manager and Media Control Platform.
  - `Outbound (2)` For the Resource Manager and Media Control Platform.
  - `Bridged (3)` For the Media Control Platform.
  - `Unknown (4)` For the Resource Manager.
  - `New Call (5)` For the Call Control Platform.
  - `Create-CCXML (6)` For the Call Control Platform.
  - `External (7)` For the Call Control Platform.
- `Local-URI` The URI that identifies the local service that was delivered.
- `Remote-URI` The URI of the party with whom the dialog was conducted. The platform obtains this information from the From header on an inbound call or the Request-URI on an outbound call.

Attributes Specific to the Media Control Platform

The Media Control Platform includes the following additional attribute in CDRs:

- For bridged calls, the parent Component ID (in other words, the Media Control Platform ID of the call session that originated the bridged session).

The Media Control Platform adds the following attributes to the CDR:

To capture usage information:

- **ASR** If ASR is used at any point during the call.
- **TTS** If TTS is used at any point during the call.
- **VOICEXML** If VoiceXML was used during the call.
- **NATIVECPA** If native media server CPD/CPA was used during the call.
- **GATEWAYCPA** If gateway-based CPD/CPA was used during the call.

To capture information about recording execution:

- **LOCALREC** If a local recording was executed during the call.
- **MSRREC** If a media stream replication recording was executed during the call.

To capture information about conference, bridging, and connection establishment:

- **CONF** If conferencing was established during the call.
- **BRIDGING** If bridging was established during the call.
- **VIDEO** If a video connection was established during the call.
- **CODEC** If any transcoding was used for this call (it does not matter which leg).

To capture MSML requests information:

- **MSPLAY** If MSML <play> was requested.
- **MSCOLLECT** If MSML <collect> or <dtmf> was requested.

CDR Attributes Specific to the Call Control Platform

The Call Control Platform includes the following additional attributes in CDRs:

How the session was started:

- **EXTERNAL** The session was created through the HTTP-session creation I/O processor.

- **CREATECCXML** <parent-ccxml-session-id> The session was created by a <createccxml> tag from a parent session.
- **NEWCALL** <call-params> The session was created because of an inbound call: <call-params> records the relevant parameters (for example, the UUID of the connection).

The reason that the CCXML session ended:

- **EXIT** The <exit> tag was executed.
- **KILL** The session was terminated by a ccxml.kill.unconditional or unhandled ccxml.kill event.
- **DOCINIT** The session ended because an error was encountered during document initialization.
- **ERROR** The session was ended by an unhandled error event.
- **SYSERR** The session ended because of an internal error.

An ID indicating the source of the session:

- For calls started by a connection, the connection ID of the initiating call.
- For externally created calls, the eventsource URI.
- For forked sessions, the Component ID of the parent session.

OR Service

The OR interface enables the Resource Manager and Media Control Platform components to accumulate statistics about call arrivals and call peaks, and it enables the Call Control Platform component to accumulate statistics about CCXML session peaks. The statistics are submitted to the Reporting Server, through the Reporting Client:

Call Arrivals

- Counts are derived from the CDRs as they are submitted or updated.
- The Resource Manager submits arrivals for the CTI and PSTN Connectors.

Call Peaks

- Statistics are derived from counts of the maximum number of concurrent calls that are observed within a given 5-minute time period.
  - The Resource Manager submits peaks for the deployment as a whole, for the CTI and PSTN Connectors individually, and for each IVR Profile that is processed on a per-tenant basis.
  - The Media Control Platform submits peaks for itself only.

The Reporting Client submits OR data to the Reporting Server at the interval that is configured in the ems.ors.reportinginterval parameter. The default is once per minute.

VAR Per-Call IVR Actions Reporting Service

The VAR Per-Call IVR Actions Reporting Service is a Reporting web service endpoint, which is available at the following URL: `/ems-rs/HIST/CDRs/MCP/VAR/actions`.

It lists the IVR actions that are occur within the lifetime of an active Media Control Platform session. Per-call IVR actions are logged with the following labels: `com.genesyslab.var.ActionStart`, `com.genesyslab.var.ActionEnd`, and `com.genesyslab.var.ActionNotes`.

This service supports `session-id` parameter, which works in conjunction with the comp-id parameter. The `session-id` parameter selects a single call that matches a session ID that is local to a given call processing server. Session IDs are not necessarily globally unique, therefore, this parameter must be accompanied by the `comp-id` parameter.

The VAR Per-Call IVR Actions Reporting also supports the `gvp-guid INSERT_TEXT` parameter. The Resource Manager manages the GUID and passes it to all call processing servers that provide service for a specific GVP session. When this parameter is specified, the report returns information for all Media Control Platform sessions that are associated with a specified GVP session (if it has been served by multiple Media Control Platform sessions).

All calls associated with a specific Genesys Management Framework session can also be selected, by using the genesys-uuid as the identifier. No two session IDs, GVP GUIDs, or Genesys UUIDs can be specified at the same time, otherwise an HTTP 400 error code is returned.

For a complete description of the Per-Call IVR Actions Report, see "Chapter 19: Historical Reports" in the Genesys Voice Platform 8.5 User's Guide.

SQA Service

The Service Quality Analysis (SQA) service on the call processing servers, provides statistics on the service quality of GVP deployments and sends this data to the Reporting Server through the Reporting Client. This differs from the analysis of operational logging and reporting, which typically focus on the availability and performance of servers and system components. Events that affect service quality might not show up in any operational logs.

Service-quality measurements account for all calls to the system. The platform itself measures the calls being handled, rather than using a sampling methodology where periodic test calls are made to the system. SQA gathers information from the platform and the applications running on the platform and prepares reports on quality at regular interval.

> **Tip**
>
> The ems.dc.enableSQA parameter can be set to false if service quality, latency, and call-failure tracking information is not required. When this parameter is disabled, the Reporting Client does not send this data to the Reporting Server.

SQA and NGI Compatibility

SQA is currently designed to work properly only with MCPs using NGI.

Some SQA metrics may show up for non-NGI configurations, because some configurations of GVP can be deployed with other interpreters (such as the Legacy GVP Interpreter). But beginning with release 8.1.5, MCP does not support those interpreters.

SQA Metrics

The Reporting Server Client obtains data about service quality from the call processing components and forwards it to the Reporting Server. The following types of service-quality data are accessible through the various SQ reporting UIs in the Monitoring > Voice Platform pane in the Genesys Administrator:

Service-Quality Calculations

The period of time for which service quality is calculated can be configured in the Media Control Platform Application. The ems.dc.serviceQualityPeriod parameter is configured with values that divide evenly into 1 hour (the default value is 15 minutes).

The following functions occur within each interval:

The Reporting Client forwards accumulated metrics to the Reporting Server, such as:

- The number of completed calls on each Media Control Platform. A call is considered complete if either an incall_end or outcall_end event is logged or the call completes abnormally (see Abnormally Terminated Calls).
- The number of failed calls on each Media Control Platform.

SQ metrics are calculated, such as:

- The percentage of successful calls on each Media platform.
- The number of completed and failed calls, as well as the percentage of successful calls for the cluster.
- Aggregated hourly, daily, weekly, and monthly summaries of the number of completed calls, number of failed calls, and percentage of successful calls. These summaries are accumulated for each Media Control Platform and for the entire cluster.
- Deletion of statistics for the basic service-quality period (15 minutes) after a configurable period of time (2 days). Hourly, daily, weekly, and monthly statistics are also deleted after a configurable period of time.
- System dialog messages that are generated if SQ data retention periods are configured in such a way that prevents statistics aggregation from being done properly.
- Logged events that contain the service-quality percentage for a specific application in the cluster. These events can be used to trigger alarms when service quality falls below the configured threshold. Logged

events are not generated when the number of completed calls in the service-quality period is less than the configured threshold.

- The percentage of successful calls for each application that has been executed for each Media Control Platform.
- The percentage of successful calls for each application that is executed in the deployment. This percentage is derived from aggregated results that are reported by the individual Media Control Platforms.

Service-Quality Failures

The SQA service provides metrics for the following failure types:

Failed Logging Sessions

When the VoiceXML interpreters on the Media Control Platform generate a <`log`> with a `com.genesyslab.quality.failure` or `com.genesyslab.quality.failure` label, the logging session is considered to have failed. SQ Failure types include data relating to latency, application, and audio gap failures (see Call Failures).

Latency Intervals

SQA uses the intervals-between-events data that is derived from component latency reporting and compares it with configured thresholds. Latency periods are measured in milliseconds(ms).

Abnormally Terminated Calls

When calls are abnormally terminated, they are marked with the Abnormal Termination failure type. The following call events are some examples of calls that are considered abnormally terminated:

- An `incall_initiated` event is logged, but no associated incall_begin or `incall_rejected` event is logged.
- An `incall_begin` event is logged, but a corresponding `incall_end` is not.
- The `incall_end` event is logged with the `syserr` reason code.
- The Resource Manager logs a `sip_session_timeout` event for the call.
- A `bridge_begin` event is logged, but no corresponding `bridge_end` event.

Call Failures

The SQA service gathers and stores call-failure information for each session that ends in each service-quality period. Call Failure Records are maintained in the database for a configured period of time (by default, 1 year).

The default value for the Call Failure Records can be overridden in the `gvp.rs.db.retention.sq.failures` configuration option. The SQA service stores the following information for each call failure:

- Session ID
- Session end time
- Application (associated with the session)
- Failure type
- Time of failure (within ms)
- Cause-of-failure description (value string)

Reporting data can be generated for various types of call failures and the SQ Failure Details report can be filtered to provide details about each of the specific call-failure types, including short strings for System Error call failures, which describe the cause-of-failure.

For more information about the configuration options that relate to all aspects of SQA, see the Genesys Voice Platform 8.5 User's Guide.

Reporting Client

The Reporting Client on each component provides reliable delivery of DATAC logs and metrics, CDRs, service-quality metrics, and OR statistics to the Reporting Server.

The Reporting Client persistently queues data when the Reporting Server is unavailable, and uses exponential back-off to attempt to reconnect to the Reporting Server. Data that is submitted to the Reporting Client is eventually sent to the Reporting Server, even if the Reporting Server is unavailable for an extended period of time due to an outage.

> **Tip**
>
> Data that is stored in memory is lost if the call-processing component shuts down unexpectedly. Data is persisted to disk only if the Reporting Client cannot successfully deliver the data to an available Reporting Server.

For improved performance, the Reporting Client can be configured to send CDRs and metrics in batches. However, this can result in slight delays in data delivery. For more information, see the description of the `ems.rc.cdr.batch_size` parameter in the Genesys Voice Platform 8.5 User's Guide.

Support for Reporting Server in TLS mode

The Reporting Client can connect to a Reporting Server in TLS mode. Whether or not it uses an encrypted connection, depends on how the activemq.connectionMode option in the messaging section of the Reporting Server (to which it is connected) is configured.

The Reporting Client **rc.truststore_certificate** configuration option in the **ems** section contains the file name of the certificate in Privacy Enhanced Mail (PEM) format. The certificate is required to connect to the Reporting Server (ActiveMQ) over TLS.

The Reporting Client **rc.keystore_certificate** and **rc.keystore_password** configuration options in the **ems** section are required to connect to Reporting Server (ActiveMQ) over Mutual TLS.

Reporting Server

As shown in the figure GVP Reporting Architecture, the services that the Reporting Server provides include the following:

- **Storage services** Logs and metrics that the Reporting Client (on the components) delivers are stored in the GVP Reporting database, where they can be queried by Reporting Web Services.
- **Reporting Web Services** HTTP web services return XML that conforms to well-defined schemas. XML-based reports are displayed on the Monitoring tab in Genesys Administrator. For more information, see Reporting Web Services.
- **Service Quality (SQ) Alarm Generator** Alarms are generated through Reporting Web Services (in Genesys Administrator) when service quality falls below configured thresholds.
- **VAR Stats Generator** The Reporting Server computes VAR statistics, based on the VAR-specific metrics that it receives (see VAR Metrics).
- **Summarization process** Every hour (on the half-hour), the Reporting Server rolls five-minute statistics into higher-level hourly, daily, weekly, and monthly summaries. The process summarizes VAR, SQ, and OR data only.

For performance reasons, the process does not start summarizing for a period until that period has ended. For example, a monthly summary for January will not be created until the start of February.

The Reporting Server can derive summaries upon request. For example, you can request a monthly report for January before January has ended.) However, this puts more load on the database than when the regular summarization process derives summaries from precomputed data.

- **Database maintenance process (DBMP)** The DBMP purges old data in accordance with data-retention policies. By default, the process runs once per day, at a configurable time. The data-retention policies are also configurable. For more information, see the section about configuring database retention policies in the Genesys Voice Platform 8.5 User's Guide.
- **Database Partitioning** The Reporting Server supports partitioning for Oracle 10g or 11g Enterprise Edition, and Microsoft SQL Server 2008 Enterprise Edition, and provides compatible schemas. Partitioning is automatically enabled during installation if either of these database editions is selected.

> ### Tip
>
> Genesys recommends that you not change the partitioning mode of operation or the number of partitions (even after the Reporting Server is started) because of issues that might arise if the database schema or stored data is changed.

- Queries the SNMP MIBs from the Supplementary Services Gateway components and provides summarized data to the Reporting Web Services.

SNMP Query and Trap Generation

To manage SNMP query and trap generation for multiple Reporting Server instances in your environment, you can configure Reporting Server connections to Net-SNMP. After the connections are established, the Reporting Server queries and generates traps in the following way:

- The Reporting Server finds the first Management Framework connection to Net-SNMP and attempts to use the agent on this connection.

- If the agent is not reachable, the Reporting Server attempts to connect to the agent at regular intervals, which is configured by using the `connection_delay_sec` option in the `agentx` section of the Reporting Server Application. The default value is 60 seconds. By using the `max_connection_attempt` configuration option in the `agentx` section, the number of times the connection is re-attempted can also be configured.

For a complete list of Reporting traps, see the GVP 8.5 SNMP MIB Reference.

Reporting Web Services

Reporting Web Services can be deployed over HTTP or HTTPS and is deployed by default, at the following URL:

`http://<Reporting Server host name>:8080/ems-rs`

In multi-tenant environments, Reporting Web Services provides the `http://<Reporting Server host name>8080/ems-rs/tenants` URL, which returns a complete list of tenants, including their names and DBIDs, that are provisioned for the environment.

When SQA is enabled in the deployment, these reports are accessible:

Report	Use this URL to access
SQ Summary	`http://<Reporting Server host name>8080/ems-rs/sqa/servicequality`
Failure Details	`http://<Reporting Server host name>8080/ems-rs/sqa/failures`
Latency	`http://<Reporting Server host name>8080/ems-rs/sqa/latency/details`
Latency Histogram	`http://<Reporting Server host name>8080/ems-rs/sqa/latency/histogram`

The reporting services return results (reports) as XML documents that conform to available Regular Language for XML Next Generation (RelaxNG) schemas, therefore, the GVP Reporting data is available to third-party reporting products, and on the Monitoring tab in the Genesys Administrator GUI. User interfaces for Service Quality Advisor (SQA) Reporting, VAR Reporting, CDR Reporting, and Operational Reporting can be used to view and filter reports and statistical data.

> Tip
>
> Browse to: `http://<Reporting Server host name>:8080/ems-rs/components` to test Reporting Server.

For detailed information about the XML schemas for GVP Reporting Web Services, contact Genesys Customer Care.

Report Categories

Reporting services are grouped into the following categories:

- Real-time
- Historical
- VAR

For more information about the GVP reports that are available in these categories, see the Monitoring part of the Genesys Voice Platform 8.5 User's Guide.